

AMENDMENTS TO THE SPECIFICATION

Please amend the following paragraph of the specification.

[0031] The KMS (24) uses three main data structures: an n-tuple ~~a-tuple~~, a vector, and a serialized file. FIG. 4 illustrates a 3-tuple in accordance with an embodiment of the present invention. The 3-tuple (47) includes three data fields: a key field (48), a value field (50), and a type field (52). The key field (48) contains an identifying name of a value, e.g., Admin_Password. The value field (50) contains a value identified by the name in the key field (48). The type field (52) may contain either "USER" or "GENERATED." "USER" corresponds to a value in the value field (50) that was entered by the user. "GENERATED" corresponds to a value in the value field (50) that was generated by the KMS, specifically the key generation tool in the encryption module. The three fields are combined to produce a 3-tuple (47). Data input into the KMS is first stored as a 3-tuple (47) within the memory prior to processing.

[0034] FIG. 6 illustrates a typical graphical user interface (GUI) to input data, in one or more embodiments of the present invention. The GUI (60) may be part of a stand alone application or integrated into a web browser. The GUI (60) provides the user with a means to input data into the KMS. A field name section (64) contains the key's e.g., card.ldap.admin.dn, which are stored in the key field of the 3-tuple. A value input section (66) contains two input text boxes: a value text box (68) and a confirm [[field]] text box (70). The user inputs a value corresponding to a key in the field name section (64) into the corresponding value text box (68), the user then re-enters the value in the confirm text box (70). In one embodiment of the present invention the text typed into the value text box (68) and the confirm text box (70) is displayed as "clear text." In another embodiment of the present invention the text typed into the value text box (68) and the confirm text box (70) is displayed as a series of asterisks. A generate randomly section (72)

contains a series of checkboxes, one for each key. The user may check a box for a given key, which prompts the KMS to generate a value for that particular key. As mentioned above, the KMS generates the values using a key generation tool within the encryption module.